TM-70-1033-7

TECHNICAL MEMORANDUM

THE COMPLETE EIGENVALUE PROBLEM FOR GENERAL REAL MATRICES

Bellcomm

BELLCOMM, INC.
955 L'ENFANT PLAZA NORTH, S.W. WASHINGTON, D.C. 20024

COVER SHEET FOR TECHNICAL MEMORANDUM

TITLE- The Complete Eigenvalue Problem for General Real Matrices

TM- 70-1033-7

DATE- June 16, 1970

FILING CASE NO(S)- 103-6, 320

AUTHOR(S)- V. Thuraisamy

FILING SUBJECT(S)-
(ASSIGNED BY AUTHOR(S)-

ABSTRACT

In this memorandum, the complete eigenvalue problem for general real matrices is considered with no assumptions as to symmetry, multiple eigenvalues or diagonalizability. A routine has been developed which finds all the eigenvalues and the eigenvectors. Eigenvectors corresponding to each eigenvalue are orthonormalized.

The problem is analyzed, and algorithms are developed, in four major steps:

1. Reduction of general matrix to Hessenberg form
2. Determination of eigenvalues of Hessenberg form
3. Construction of eigenvectors of Hessenberg form using inverse iteration
4. Construction of an orthnormal eigenvector basis for eigenspace of each eigenvalue of original matrix.

A computer program based on these algorithms has been implemented. Some test results are included. An elementary proof of convergence of the inverse iteration process for defective matrices is given.

SEE REVERSE SIDE FOR DISTRIBUTION LIST

DISTRIBUTION

COMPLETE MEMORANDUM TO

CORRESPONDENCE FILES:

OFFICIAL FILE COPY
plus one white copy for each
additional case referenced

TECHNICAL LIBRARY (4)

NASA Headquarters
T. A. Keegan - MA-2

MSC
D. C. Wade - ES2

MSFC
J. B. Sterrett - S&E-ASTN-A

Bellcomm, Inc.
A. P. Boysen, Jr.
F. A. Brewer
D. R. Hagner
H. A. Helm
S. N. Hou
B. T. Howard
S. Kaufman
P. F. Long
D. L. Mather
R. K. McFarland
J. Z. Menard
L. D. Nelson
J. M. Nervik
I. M. Ross
R. V. Sperry
H. E. Stephens
J. W. Timko
R. L. Wagner
M. P. Wilson
All Members of Department 1033
Central File
Department 1024 File

COVER SHEET ONLY TO

BELLCOMM, INC.
955 L'ENFANT PLAZA NORTH, S.W. WASHINGTON, D.C. 20024

SUBJECT: The Complete Eigenvalue Problem for General Real Matrices - Cases 103-6, 320

DATE: June 16, 1970

FROM: V. Thuraisamy

TM-70-1033-7

TECHNICAL MEMORANDUM

INTRODUCTION

A pervasive problem in applied mathematics is the algebraic eigenvalue problem. Existing programs for the solution of eigenvalue problems are inadequate to meet the needs of increasingly sophisticated technical analysis. This memorandum contains a detailed analysis of the eigenvalues and eigenvectors of general real matrices.

A number of convenient properties possessed by the symmetric matrix are no longer available for the general matrix. The two most serious difficulties are the possible nondiagonalizability and the fact that eigenvectors of distinct eigenvalues are not necessarily orthogonal. As a consequence of the latter, we do not have a basis of orthonormal eigenvectors. There are several methods available for solving the symmetric eigenvalue problem. The "Q-R" method, is efficient for finding the eigenvalues of nonsymmetric matrices as well. But there does not appear to be any program available for extracting the eigenvectors of repeated eigenvalues in the nonsymmetric case. Moreover, if the matrix is nondiagonalizable then there are less eigenvectors than eigenvalues; thus at least one of the repeated eigenvalues will have a higher multiplicity than the dimension of the corresponding eigenspace. These are some of the difficulties the herein contained algorithms are designed to handle.

A computer program based on these algorithms has been written and tested on several matrices collectively known to possess the above mentioned difficulties. Results of some of these test examples are presented in Section 5. A companion memorandum documenting the computer program will be forthcoming. The program is written to produce a complete set of eigenvectors, although it can be easily modified to find the eigenvectors of selected eigenvalues. For example, in structural analysis, the eigenvectors of a few eigenvalues at one (usually lower) end of the spectrum are much more valuable than the rest. Again, for large systems it may be impractical and/or unnecessary to insist on computing all the eigenvectors.

0. Preliminaries

In general a matrix is indicated by an upper case letter and a vector by a lower case letter.

A is the given n x n matrix with elements a_{ij}. An n-dimensional vector v with components $\{v_1, v_2, \ldots, v_n\}$ has norm

$$||v|| = \sqrt{\left(\sum_{i=1}^{n} |v_i|^2\right)} , \qquad (1)$$

where $|x|$ is the absolute value of the real or complex number x. If $||v|| = 1$, v is called a unit vector. Since several vectors will be made, it must be understood that a quantity like v_{ir} may refer to the i^{th} component of the vector v_r. For convenience of writing, a vector may be represented by its transpose thus: $v^T = \{v_1, \ldots, v_n\}$. A column of a matrix will sometimes be considered as a vector. x* denotes the complex conjugate of the number x. M^T is the transposed matrix of a (possibly rectangular) matrix M. Thus v^T is a row vector. The notation (,) will be used for the inner product of vectors:

$$(u,v) = \sum_{i=1}^{n} u_i v_i^* \qquad (2)$$

Thus, in particular, $(u,u) = u^T u = ||u||^2$.

Remark

The so-called 2-norm is chosen here. Associated with a vector norm matrix norms are defined by

$$||Ax|| \leq ||A|| \; ||x|| \quad . \qquad (3)$$

Any matrix or vector norm for which (3) holds is said to be compatible or consistent. Two popular matrix norms compatible with the 2-norm (1) are

$$||A||_2 = (\text{max eigval of } A^H A)^{1/2} \tag{4a}$$

and

$$||A||_E = \left(\sum_i \sum_j |a_{ij}|^2\right)^{1/2} \tag{4b}$$

In (4) A^H is the complex conjugate transpose of A. $||A||_2$ is called the spectral norm and $||A||_E$ the Euclidean norm. Even though the Euclidean norm is weaker than the spectral norm, because it is readily computable, it is useful in error analysis.

Definition

An n x n matrix H is in upper Hessenberg form if and only if

$$h_{ij} = 0 \text{ for all } i > j+1 \ (i, j=1,2,\ldots, n)$$

The reader is reminded that a nondiagonalizable matrix is also called a defective matrix because its eigenspace is of a lower dimension than that of the matrix. Equivalently it is said to possess nonlinear divisors. A simple example of a nondiagonalizable matrix is the following:

$$C = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The eigenvalues of the 3 x 3 matrix are 1, 1 and 1. {1, 0, 0} and {0, 0, 1} are the eigenvectors. Also there are no other independent eigenvectors. Thus the eigenspace does not span the three dimensional vector space on which the matrix C operates as a linear operator (linear transformation).

When a matrix is brought into Jordan-canonical form, not only are the eigenvalues known but a complete description is available as to the multiplicity of each eigenvalue, the number of eigenvectors for each eigenvalue and so on. The above matrix was deliberately chosen to be already in Jordan-canonical form from which the following can be stated. C has eigenvalue 1 of multiplicity three. There is one 2 x 2 so-called Jordan matrix (the leading principal submatrix here) and therefore there is exactly one eigenvector for two of these eigenvalues. The third eigenvalue corresponding to the 1 x 1 Jordan matrix will have its own independent eigenvector (namely, {0, 0, 1}). The elementary divisors are the characteristic polynomials corresponding to each of the Jordan submatrices. For C, they are

$$(x-1)^2 \ , \quad (x-1) \ .$$

Thus C has a nonlinear elementary divisor. Therefore it is defective and nondiagonalizable.

Unfortunately, it is not feasible in general to obtain the Jordan-canonical form. Consequently, aposteriori decisions are made as to defectiveness, etc., in the sense that these decisions are delayed till after the eigenvalues and eigenvectors are found. These questions are discussed in section 3. Some of the features of the Jordan-canonical form are exhibited in the following example.

$$E = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix} \tag{5}$$

While obtaining the Jordan form is a major undertaking, the det(E-λI) can be written as

$$p(\lambda) = (-1-\lambda)\{(-1-\lambda)[(-1-\lambda)(-2-\lambda)]\}$$

$$= (\lambda+1)^3(\lambda+2)$$

Therefore, the eigenvalues are -1, -1, -1, -2. The characteristic polynomial is

$$p(x) = (x+1)^3(x+2)$$

By definition the minimum polynomial m(x) for matrix E is the monic polynomial with lowest degree such that

$$m(E) = 0 \ .$$

The Cayley-Hamilton theorem states that

$$p(E) = 0$$

I.e.,

$$(E+I)^3(E+2I) = 0$$

Hence m(x) divides p(x). For E it can be verified that $m(x) \equiv p(x)$. This implies that there is only one eigenvector for the repeated eigenvalue -1. Thus E is a defective matrix and has the Jordan-canonical form

$$\begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -2 \end{bmatrix}$$

As exemplified above, one can extract from the minimum polynomial much information including whether or not defects are present. But in general it is not feasible to find this polynomial numerically. Section 5 includes numerical results for E.

The herein discussed program consists of the following subprograms:

1. The given matrix A is condensed by Householder similarity transformations to an upper Hessenberg form H.

2. The eigenvalues of H are computed by the 'Q-R' method.

3. The eigenvectors of H are computed using inverse iteration and for every eigenvalue an orthonormal set as basis for its eigenspace is derived by the Gram-Schmidt process.

4. The eigenvectors of A are now obtained from those of H. Again the eigenspace of each eigenvalue is spanned by an orthonormal set of eigenvectors of A. The eigenvalues of A are, of course, the same as those of H.

There are two reasons why the problem is restricted to real matrices. One is that it appears the current need is only for real matrices. The other reason is that the extra demand for storage caused by including complex matrices would necessarily reduce the maximum dimension of the matrix that could be handled in-core.

It is emphasized that A is only assumed to be real and that there is

a. no restriction on symmetry

b. no restriction on repeated eigenvalues

c. no restriction on diagonalizability.

The various subprograms are now described. The details are given only as far as necessary to understand and appreciate the underlying principles employed in the program. The methods of sections 2 and 3 do not require a Hessenberg form but if these are applied to a full matrix, the amount of computation would soon be prohibitively large.

1. Matrix to Condensed Form

In a recent paper [1] Businger has exhibited an example to prove that the reduction of a general matrix to Hessenberg form by elementary similarity transformation using the popular Gaussian elimination may be unstable. Therefore, Householder reduction is employed to transform A to a similar upper Hessenberg form H. This method is known to be unconditionally stable. (Householder's unitary matrices have been used to tridiagonalize symmetric matrices in the subroutine TRDMX available in Bellcomm's computer library.) The similarity transformation from A to H is accomplished by the use of (n-2) elementary unitary matrices P_r; r=1, 2, ..., n-2. The P_r have the simple form

$$P_r = I - 2w_r w_r^T \qquad (6)$$

where I is the n x n identity matrix and w_r is a real unit vector with the first r components zero. Writing

$$A_0 = A$$

$$A_r = P_r A_{r-1} P_r \quad (r=1,2,\ldots,n-2) \qquad (7)$$

it turns out that with proper choice of the w_r, A_{n-2} is in fact an upper Hessenberg matrix. That is to say, $H = A_{n-2}$. Consequently, the method of choosing the w_r is explained.

Method

The first step is to choose w_1 such that all the last (n-2) elements of the first column of A_1 (in (7)) are zero.

Since w_1 and P_1 have the forms[1]

$$w_1^T = \{0, x_2, \ldots, x_n\} \quad , \tag{8}$$

$$P_1 = \begin{bmatrix} 1 & 0 & & \cdots & 0 \\ 0 & 1-2x_2^2 & -2x_2x_3 & \cdots & -2x_2x_n \\ \vdots & -2x_3x_2 & & & \\ \vdots & \vdots & & & \\ 0 & -2x_nx_2 & & & 1-2x_n^2 \end{bmatrix} \quad , \tag{9}$$

it is clear that post multiplication by P_1 leaves the first column of any matrix unaltered. Hence $P_1A_0P_1$ will have the last (n-2) elements of column one vanish if and only if (P_1A_0) also has the last (n-2) elements of the first column vanish. Now

$$P_1A_0 = (I-2w_1w_1^T)A$$

$$= A - 2w_1w_1^TA$$

and if w_1^TA is written as t^T, then

$$P_1A_0 = A - 2w_1t^T \quad .$$

[1] The notation and approach given in [2] for symmetric matrices is used.

The elements in the first column of P_1A_0 are

$$\{a_{11}, a_{21} - 2x_2t_1, a_{31} - 2x_3t_1, \ldots, a_{n1} - 2x_nt_1\} \qquad (10)$$

where the first component t_1 of the vector t is given by

$$t_1 = x_2a_{21} + x_3a_{31} + \ldots + x_na_{n1} \quad . \qquad (11)$$

Therefore,

$$a_{i1} - 2x_it_1 = 0 \ , \ i=3,4,\ldots,n \qquad (12)$$

and also the second component $a_{21} - 2x_2t_1$ in (10) must be such that

$$(a_{21} - 2x_2t_1)^2 = \sum_{i=2}^{n} a_{i1}{}^2 \qquad (13)$$

To justify (13) it is noted that (since $w^Tw = (w,w) = 1$)

$$\begin{aligned} PP^T &= (I - 2ww^T)(I - 2ww^T) \\ &= I - 4ww^T + 4w(w^Tw)w^T \\ &= I \end{aligned}$$

Hence $P^T = P^{-1}$ (i.e., P is orthogonal). Consequently, for any vector u,

$$\begin{aligned} (Pu, Pu) &= (u, P^TPu) \\ &= (u, u) \end{aligned}$$

which implies that $||u||$ is invariant under P. Letting u be the first column of A, (13) follows from (12) and (10).

Now let the right side of (13) be s and rewrite the equation in the form

$$a_{21}x_2 - 2x_2^2 t_1 = \pm \sqrt{s}\; x_2 \tag{14}$$

Rewrite (12) in the form

$$a_{i1}x_i - 2x_i^2 t_1 = 0, \quad i=3,4,\ldots,n. \tag{15}$$

From (11), (14) and (15) it follows that

$$t_1 - 2t_1 \sum_{i=2}^{n} x_i^2 = \pm\, x_2 \sqrt{s} \;. \tag{16}$$

But (8) implies

$$\sum_{i=2}^{n} x_i^2 = 1 \;. \tag{17}$$

Hence

$$t_1 = \mp\, x_2 \sqrt{s}$$

and (12) and (13) give, respectively,

$$x_i = \mp \frac{a_{i1}}{2x_2 \sqrt{s}} \;, \quad i=3,4,\ldots,n \;.$$

$$x_2^2 = \frac{1}{2}\left[1 \mp \frac{a_{21}}{s^{1/2}}\right] \tag{18}$$

Thus w_1 and, therefore, P_1 are completely determined when $\sqrt{s}$ and then the x_i are computed from (18). The ambiguity of sign in (18) is removed by assigning the appropriate sign which will avoid cancellation. Thus they are given the same sign as a_{21}. It is also noted that

$$A_1 = P_1 A_0 P_1 = (I-2ww^T) A (I-2ww^T)$$

$$= A - 2Aww^T - 2ww^TA + 4ww^TAww^T$$

$$= A - 2w[w^TA - (w^TAw)w^T]$$

$$- 2[Aw - w(w^TAw)]w^T .$$

w^TAw is a scalar which is denoted by k. Then

$$A_1 = A - 2w\, q^T - 2q'w^T \tag{19}$$

where

$$q^T = [w^TA - kw^T] \tag{20}$$

$$q' = [Aw - kw] \ . \tag{21}$$

Thus in practice P_1 is not computed explicitly. Once the vector w is obtained the scalar k and the vectors q and q' are calculated. (19) will then furnish A_1. This results in great saving in storage since one can work with one matrix and some vectors rather than two matrices. Also there is considerable reduction in computation time since matrix multiplications are replaced by matrix-vector or vector-vector multiplications.

This completes the discussion on obtaining a matrix A_1 similar to A but with all its last (n-2) elements in the first column equal to zero. Now it is clear that one can repeat the process in going from A_1 to A_2. That is to say, find the w_2 such that the second column of A_2 would be nonzero only in the leading three places. What has to be guaranteed, however, is that this new transformation leaves the zeros of column one intact. This is shown below.

By definition w_2 is of the form

$$w_2^T = \{0,\ 0,\ x_3,\ x_4,\ \ldots,\ x_n\}$$

and P_2 has the form

$$P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1-2x_3^2 & -2x_3x_4 & \cdots & -2x_3x_n \\ 0 & 0 & -2x_4x_3 & \cdot & & \\ \vdots & \vdots & & & \cdot & \\ 0 & 0 & & & \cdot & 1-2x_n^2 \end{bmatrix} .$$

Clearly then, any matrix post-multiplied by P_2 will have its first two columns unchanged. Thus one need consider only the first column of P_2A_1. Since A_1 has the form

$$A_1 = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ 0 & & & \\ \vdots & & & \\ 0 & & & a_{nn} \end{bmatrix}$$

it is evident that the first column of P_2A_1 is identical with the first column of A_1.

In general the transformation from A_r to A_{r+1} is such that the first r columns of A_r and A_{r+1} are identical. For efficiency the computer program has taken advantage of this fact.

To summarize: There are (n-2) major steps in reducing the given matrix A to a Hessenberg matrix H. At each step a scalar s_r and a vector w_r are computed. This involves computing two square roots, once for $\sqrt{s}$ and again for obtaining the first nonzero component of w from an equation of the form (18). Once w_r is obtained it is a simple matter to compute the scalar k_r and the vectors q_r, q_r' and finally a new matrix A_r. At each successive step the number of operations is reduced and as a rough estimate about $\frac{4}{3} n^3$ multiplications, $\frac{1}{2} n^2$ divisions and 2n square roots are needed.

To fix the ideas the computation[2] will be carried out completely for the following 4 x 4 matrix.

Consider again the matrix E of (5)

$$A_0 = E = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix}$$

The aim is to show that the algorithm produces a similar Hessenberg form. With the notation of the text;

$$s_1 = \sqrt{a_{21}^{\ 2} + a_{31}^{\ 2} + a_{41}^{\ 2}} = \sqrt{3} = 1.732$$

$$w_1^T = \{0, x_2, x_3, x_4\} \quad \text{where}$$

$$x_2^{\ 2} = \frac{1}{2} + \frac{1}{2} \frac{|a_{21}|}{s_1}, \quad x_3 = \frac{a_{31} \text{sign } a_{21}}{2x_2 s_1}, \quad x_4 = \frac{a_{41} \text{sign } a_{21}}{2x_2 s_1}$$

[2] The computation is rounded to three decimals.

Hence

$$x_2 = .888$$

$$x_3 = .325$$

$$x_4 = .325$$

$$w^T A = \{1.538, -.888, .888, .238\}$$

$$(Aw)^T = \{0, -.238, -.325, -.325\}$$

$$k = w^T Aw = -.423$$

$$q'^T = [Aw - kw]^T = \{0, .137, -.188, -.188\}$$

$$q^T = [w^T A - kw^T] = \{1.538, -.513, 1.025, .375\} \quad .$$

$A_1 = A - 2w_1 q^T - 2q' w_1^T$, becomes

$$A_1 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ -1.732 & -.333 & -.911 & .244 \\ 0 & .667 & -1.544 & -.122 \\ 0 & .667 & .455 & -2.122 \end{bmatrix}$$

The elements of A_1 are correct to three decimal places. The second and final step is to reduce the (4,2) element to zero.

$$s_2 = \sqrt{a_{32}^2 + a_{42}^2} = \sqrt{.667^2 + .667^2} = .667\sqrt{2} = .943$$

$$w_2^T = \{0, 0, .924, .383\}$$

$$w_2^T A = \{.871, 1.000, -1.253, -.925\}$$

$$(Aw)^T = \{0, -.748, -1.474, -.391\}$$

$k = -1.511$

$q'^T = \{0, -.748, -7.746, .187\}$

$q^T = \{.871, 1.000, .143, -.346\}$

A_2 now becomes

$$A_2 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1.732 & -.333 & .471 & .816 \\ 0 & -.942 & -1.667 & .577 \\ 0 & 0 & 0 & -2.000 \end{bmatrix}$$

Thus A_2 is in Hessenberg form. The (4,3) element of A_2 in this example happens to be zero to our working accuracy. This is, of course, not true in general.

2. Eigenvalues of H

To obtain eigenvalues of H the so-called Q-R method is used. For Hessenberg matrices this is particularly efficient. This is an iterative method usually with rapid convergence. The method shall not be described here for three reasons. One is that it would take considerable space to present the details. Second reason is that in the program a Q-R routine developed elsewhere is used. Finally, there is an excellent exposition of the Q-R method in a recent paper by Martin, Peters and Wilkinson [3].

The idea is to transform H to an almost upper triangular form by orthogonal similarity transformations. Then the eigenvalues are simply the diagonal terms of the latter or the eigenvalues of certain 2 x 2 matrices. The basic algorithm is defined by

$$\left.\begin{aligned} H_s &= Q_s R_s \\ H_{s+1} &= Q_s^T H_s Q_s = R_s Q_s \end{aligned}\right\} \quad s = 1,2,\ldots \qquad (22)$$

where Q is orthogonal and R is upper triangular. H_1 is the initial Hessenberg matrix. It turns out that all H_s are then Hessenberg and it can be shown that $Q_s \rightarrow D$, a diagonal matrix, as $s \rightarrow \infty$, so that

H_s itself approaches an almost upper triangular matrix, provided the eigenvalues of H are suitably restricted (i.e., H is scaled).

The original Q-R method as developed independently by Francis and Kublanovskaya in 1961 has since been modified and improved by several authors and in the current form is quite sophisticated [3]. One merely observes that transformations of the type (6) are commonly used in this process as well. To speed up convergence, (22) are modified by so-called origin shifts. To avoid complex arithmetic when working with real matrices, two steps of the Q-R algorithm are performed simultaneously. As the subdiagonal elements of successive H_s rapidly approach zero, various criteria have been devised to split the matrix into two or more submatrices and the process continued with each submatrix separately until finally all the eigenvalues are isolated on the diagonal or are the eigenvalues of 2 x 2 matrices.

3. Eigenvectors of H

It is now assumed that reasonably accurate eigenvalues have been obtained for H. The next thing to do is to compute the eigenvectors of H. Obviously there are at least as many methods of doing this as there are for solving a system of linear equations. But none can guarantee a complete set of independent vectors when there are repeated roots. Most methods would in fact give only one eigenvector for each distinct eigenvalue. A method is described here that determines the complete eigenspace for each distinct eigenvalue. Moreover, the eigenvectors are orthonormal. This method can also be used to compute eigenvectors for selected eigenvalues. The program has been tested on some examples with repeated eigenvalues and the results were satisfactory (see section 5). The tests included matrices with nonlinear elementary divisors.

A discussion of the question of determining the multiplicity of an eigenvalue follows. When the matrix has relatively poorly separated eigenvalues, this determination can be difficult. It is important to know the multiplicity of an eigenvalue; if $\lambda_1 \neq \lambda_2$ then there is at least one eigenvector for each of these eigenvalues, whereas if $\lambda_1 = \lambda_2$, then there may exist only one eigenvector. Mistaking equal eigenvalues to be distinct is a bigger hazard than taking close eigenvalues as equal, because in the former, one may be finding an eigenvector which does not exist! Such difficulties have led to the belief that some sort of ordering of the complex numbers is needed. Attention is now directed to a description of how these difficulties can be resolved.

Let ϵ_1, ϵ_2 and ϵ_3 be small positive numbers. The actual values assigned to these numbers are pretty much left to the user. Let the eigenvalues be $\lambda_j = \mu_j + i\nu_j$, $j=1,\ldots,n$. First test for pure real and pure imaginary numbers.

a. If $|\lambda_j| \leq \epsilon_1$, set $\lambda_j \equiv 0$.

b. If $|\nu_j|/|\mu_j| \leq \epsilon_2$ or if $|\nu_j| \leq \epsilon_2$ set $\nu_j=0$.

c. If $|\nu_j|/|\mu_j| > \epsilon_2$ and if $|\mu_j|/|\nu_j| \leq \epsilon_2$ or if $|\mu_j| \leq \epsilon_2$ set $\mu_j=0$.

The following four steps are used to test for multiplicity. i) First reorder all eigenvalues in ascending order of real parts. ii) Then divide eigenvalues into classes as follows. Two neighboring eigenvalues λ_j and λ_{j+1} would be in the same class if and only if $\mu_{j+1} - \mu_j \leq \epsilon_3$. Any two eigenvalues λ_j and λ_k $(j<k)$ would be in the same class if and only if λ_ℓ lies in the same class as $\lambda_{\ell-1}$ for all $j < \ell \leq k$. Thus the number of classes could be anything between 1 and n depending on ϵ_3 and the distribution of the spectrum. iii) For each class arrange the members according to increasing order of imaginary parts. iv) Finally, divide eigenvalues in each class into subclasses (cells) where two eigenvalues λ_j and λ_{j+1} would be in the same subclass if and only if $\nu_{j+1} - \nu_j \leq \epsilon_3$. As in (ii) above, λ_{j+2} is in the same class as λ_{j+1} if and only if $\nu_{j+2} - \nu_{j+1} < \epsilon_3$ and so on.

Thus the n eigenvalues have been grouped into, say, m nonempty sets so that the computed eigenvalues corresponding to each repeated root must fall in the same set for appropriate ϵ_3. For most problems, a judicious choice of ϵ_3 should give a grouping such that each set has precisely the repeated roots. The m would be exactly the number of distinct eigenvalues and there must be at least m independent eigenvectors.

Grouping the eigenvalues in the above fashion one can, by taking ϵ_3 sufficiently large, make sure that all the computed eigenvalues corresponding to a multiple root fall in the same cell. If a multiple root is also close to one or more other roots, then our cell may conceivably include one or more of these other roots as well. But now one can at least intensify tests within each cell to separate the distinct eigenvalues. The reader, however, should be warned of the significance of situations where an eigenvalue is pathologically close to a repeated eigenvalue. For example, consider the matrix

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & a \end{bmatrix}$$

The eigenvalues are -1, -1, -1 and a. The repeated eigenvalue has unique eigenvector $\{0, 1, 0, 0\}$ and the eigenvalue a has the eigenvector $\{0, 1, 0, 1+a\}$ when $a \neq -1$. Thus if $a+1 = \varepsilon$ and ε is down to the noise level, one cannot say, in practice, whether the matrix has -1 as a root of multiplicity four and a unique eigenvector or has two distinct but very close roots and therefore two distinct eigenvectors. It appears that a pathologically close eigenvalue mistakenly identified as a repeated root does no harm even though this in general means dropping a genuinely existing eigenvector from the list or orthonormalizing an eigenvector which should not be orthogonalized.

The proposals exemplified in the last few paragraphs, specifically designed to handle multiplicities and defects, have not yet been incorporated into the computer program. An obvious difficulty is to come up with satisfactory criteria for the parameters ε_i. As the program stands now, multiplicities and defects are discerned aposteriori, in the following sense. Two computed values are treated as if they are distinct unless they turn out to be identical to machine accuracy. Thus in practice there seldom are repeated roots and consequently one will end up with as many eigenvectors as the dimension of the matrix. But examining the computed eigenvalues and eigenvectors one is usually able to determine multiplicities as well as defects if any (see section 5).

A proof for convergence of the inverse iteration method follows. The proof has been indicated for symmetric matrices in Wilkinson [4] and convergence has been tacitly assumed for general matrices, but a proof for general nonsymmetric matrices which may be defective appears to be unavailable.

First assume that H (or A, equivalently) is not defective. Let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the eigenvalues and $u_1, u_2, \ldots, u_n$ be the corresponding set of eigenvectors. Then an arbitrary vector x_0' has a representation

$$x_0' = \sum_{i=1}^{n} b_i u_i \tag{23}$$

for scalars b_i. If λ is not one of the eigenvalues λ_k, then

$$(H - \lambda I)y_1 = x_0' \tag{24}$$

has the unique solution

$$y_1 = \sum_{i=1}^{n} (H - \lambda I)^{-1} b_i u_i \quad . \tag{25}$$

For a specific i_0, the corresponding vector in the sum (25) has the representation (for some scalars e_k)

$$(H - \lambda I)^{-1} b_{i_0} u_{i_0} = \sum_{k=1}^{n} e_k^{(i_0)} u_k \tag{26}$$

so that

$$b_{i_0} u_{i_0} = (H - \lambda I) \sum_{k=1}^{n} e_k^{(i_0)} u_k$$

$$= \sum_{k=1}^{n} (H - \lambda_k I) e_k^{(i_0)} u_k + \sum_{k=1}^{n} (\lambda_k - \lambda) e_k^{(i_0)} u_k \quad .$$

I.e.,

$$b_{i_0} u_{i_0} = \sum_{k=1}^{n} e_k^{(i_0)} (\lambda_k - \lambda) u_k \tag{27}$$

since (λ_k, u_k) are eigen pairs of H. But the u_k are a linearly independent set. Hence

$$
\begin{aligned}
e_{i_0}^{(i_0)}(\lambda_{i_0} - \lambda) &= b_{i_0} \\
e_k^{(i_0)} &= 0 \ , \ k \neq i_0 \ .
\end{aligned}
\tag{28}
$$

Thus (26) and (25) now give respectively

$$(H - \lambda I)^{-1} b_{i_0} u_{i_0} = \frac{b_{i_0} u_{i_0}}{\lambda_{i_0} - \lambda}$$

and[†]

$$y_1 = \sum_{i=1}^{n} \frac{b_i u_i}{\lambda_i - \lambda} \tag{29}$$

Hence if λ is an approximation to a simple eigenvalue λ_j, x_1 may be taken to be the normalized vector

$$x_1 = \sum_{k=1}^{n} \beta_k^{(1)} u_k$$

where

$$\beta_j^{(1)} = 1 \quad \text{(normalization)}$$

$$|\beta_k^{(1)}| = \frac{|b_k||\lambda_j - \lambda|}{|\lambda_k - \lambda||b_j|} \ , \quad k \neq j \ .$$

[†] It was pointed out by Professor J. S. Vandergraft that equation (29) can be obtained more easily using the fact that $(H-\lambda I)^{-1} u_i = \frac{1}{\lambda_i - \lambda} u_i$. But this approach is followed because equations of type (23) - (26) appear necessary for the defective case.

If one writes

$$(H - \lambda I)\ y_{i+1} = x_i,\ x_{i+1} = \text{normalized } y_{i+1}, \qquad i=0,\ 1,\ 2,\ \ldots \tag{30}$$

then (24) represents the first step in this iteration. In general, therefore, if the new vectors are normalized before being fed into the right side of (30), the coefficients in

$$x_s = \sum_k \beta_k^{(s)} u_k \tag{31}$$

will satisfy the relation

$$\beta_j^{(s)} = 1 \quad \text{(normalization)}$$

$$\left|\beta_k^{(s)}\right| = \left|\frac{b_k}{b_j}\right| \cdot \left|\frac{\lambda_j - \lambda}{\lambda_k - \lambda}\right|^s, \quad k \neq j\ . \tag{32}$$

It is clear from (32) that if λ_j is a "well-separated" simple eigenvalue, then for $k \neq j$

$$\beta_k^{(s)} \to 0 \quad \text{as } s \to \infty$$

rather rapidly, provided $b_j \neq 0$.

This in turn implies (from (31)) that $x_s \to u_j$, the eigenvector for λ_j.

Now suppose that λ_j is a repeated root. For definiteness say that $\lambda_1 = \lambda_2 > \lambda_3 \geq \ldots \geq \lambda_n$. I.e., let the largest eigenvalue λ_1 be of multiplicity two.

If λ is now an approximation to λ_1, then (31) may be written

$$x_s' = \beta_1^{(s)}u_1 + \beta_2^{(s)}u_2 + \sum_{k=3}^{n} \beta_k^{(s)}u_k \tag{33}$$

and if one normalizes as before, namely,

$$\beta_1^{(s)} = 1$$

then similar to equations (32),

$$|\beta_k^{(s)}| = \left|\frac{b_k}{b_1}\right| \left|\frac{\lambda_1-\lambda}{\lambda_k-\lambda}\right|^s \quad , \; k \neq 1 \; . \tag{34}$$

Consequently, for $k \geq 3$, $\beta_k^{(s)} \to 0$ as $s \to \infty$. But for k=2,

$$|\beta_2^{(s)}| = \left|\frac{b_2}{b_1}\right|$$

Hence, in general, $\beta_2^{(s)} \to \beta_2 \neq 0$. Therefore, with this normalization, (33) implies that

$$x_s' \to u_1 + \beta_2 u_2 \tag{35}$$

Thus x_s' tends to a vector in the eigenspace of λ_1; i.e., the vector space spanned by all the eigenvectors of λ_1. Since every vector in this eigenspace is an eigenvector for λ_1, one in fact has, in this limiting vector (35), an eigenvector for λ_1. This argument clearly can be extended to any eigenvalue of finite multiplicity.

Now look at the case when H is defective. Use the following result from linear algebra.

Theorem 1. (Primary decomposition)

Let $c_1, c_2, \ldots, c_r$ be the r distinct eigenvalues of H and the minimum polynomial of H be

$$m(x) = (x-c_1)^{s_1}(x-c_2)^{s_2} \ldots (x-c_r)^{s_r} .$$

Then

a. the n-dimensional space V over the complex field has direct sum

$$V = W_1 \oplus W_2 \oplus \ldots \oplus W_r$$

and

b. each W_i is invariant under H where

$$W_i = \text{nullspace of } (H-c_iI)^{s_i} , \ i=1,2,\ldots,r.$$

A proof of this theorem may be found, e.g., in Hoffman and Kunze [6]. It is clear that every eigenvector v_i for the eigenvalue c_i is in W_i. In fact the dimension of W_i is the same as the multiplicity of the eigenvalue c_i. If all the s_i are one, then H is diagonalizable and therefore H is not defective. Hence take at least one $s_i > 1$. The special case of diagonalizable matrices was considered separately because the proof was simple. The next several paragraphs are about defective systems.

A detailed proof of the convergence of the iterative scheme for defective systems is given in the Appendix. Only a specific example is considered here which brings out all the essential features of the proof. The 4 x 4 matrix E given in (5) is selected.

$$E = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix}$$

The eigenvalues of this matrix were shown to be -1, -1, -1 and -2. For this matrix, the minimum polynomial m(x) is the same as the characteristic polynomial p(x):

$$p(x) = m(x) = (x+1)^3(x+2)$$

$$(E+I) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & -1 \end{bmatrix}$$

Solving $(E+I)x = 0$ yields $x^T = \{0, 1, 0, 0\}$ as the only independent eigenvector for the eigenvalue -1. The eigenvector for the eigenvalue -2 is $\{0, -1, 0, 1\}$.

$$(E+I)^3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

and the vectors

$$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} , \quad \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} , \quad \begin{bmatrix} 0 \\ 0 \\ 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

are a basis for the nullspace of $(E+I)^3$ and, of course, the eigenvector $\{0, -1\sqrt{2}, 0, 1/\sqrt{2}\}$ is a basis for the nullspace of $(E+2I)$. These four vectors (noting they are not all eigenvectors) are called, respectively, u_1, u_2, u_3 and u_4. Let λ be the computed approximation to the eigenvalue -1 and let $\varepsilon = -1-\lambda$ be the error. One can repeat equations (23 - (26) here but must replace (27) by

$$b_{i_0} u_{i_0} = \sum_{2}^{4} (E-\lambda I) e_k^{(i_0)} u_k + \varepsilon e_1^{(i_0)} u_1 \quad .$$

The aim is to evaluate $e_k^{(i_0)}$, $k=1,2,3,4$; $i_0=1,2,3,4$.

Let $i_0=1$. Then

$$Eu_2 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = u_1 - u_2 + \sqrt{2}\, u_3$$

$$Eu_3 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0 \\ \sqrt{2} \\ -1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} = \sqrt{2}\, u_1 - u_3$$

$$Eu_4 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} 0 \\ -1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0 \\ \sqrt{2} \\ 0 \\ -\sqrt{2} \end{bmatrix} = -2u_4$$

(36) can now be rewritten

$$\begin{aligned} b_1u_1 = {} & e_2^{(1)}(u_1-u_2+\sqrt{2}\ u_3) + e_3^{(1)}(\sqrt{2}\ u_1-u_3) \\ & + e_4^{(1)}(-2u_4) - \lambda(e_2^{(1)}u_2 + e_3^{(1)}u_3 + e_4^{(1)}u_4) \qquad (37) \\ & + \varepsilon e_1^{(1)}u_1 \end{aligned}$$

Since the u_i are linearly independent, one may equate coefficients:

u_1: $b_1 = e_2^{(1)} + \sqrt{2}e_3^{(1)} + \varepsilon e_1^{(1)}$

u_2: $0 = -e_2^{(1)} - \lambda e_2^{(1)}$

u_3: $0 = -e_3^{(1)} + \sqrt{2}\ e_2^{(1)} - \lambda e_3^{(1)}$

u_4: $0 = e_4^{(1)}(-2-\lambda)$.

Hence

$$e_1^{(1)} = \frac{b_1}{\varepsilon} \quad , \quad e_2^{(1)} = 0 = e_3^{(1)} = e_4^{(1)} \quad .$$

If $i_0 = 2$, then corresponding to (37) one has

$0 = \varepsilon e_1^{(2)} + e_2^{(2)} + \sqrt{2}\ e_3^{(2)}$

$b_2 = -e_2^{(2)} - \lambda e_2^{(2)}$

$$0 = -e_3^{(2)} + \sqrt{2}\, e_2^{(2)} - \lambda e_3^{(2)}$$

$$0 = e_4^{(2)}(-2-\lambda) \ .$$

Thus

$$e_2^{(2)} = \frac{b_2}{-(1+\lambda)} = \frac{b_2}{\varepsilon}$$

$$e_3^{(2)} = \frac{\sqrt{2}\, e_2^{(2)}}{1+\lambda} = \frac{-\sqrt{2}\, b_2}{\varepsilon^2}$$

$$e_1^{(2)} = (-e_2^{(2)} - \sqrt{2}\, e_3^{(2)})/\varepsilon = \frac{b_2(2-\varepsilon)}{\varepsilon^3}$$

$$e_4^{(2)} = 0.$$

Again

$$e_4^{(3)} = 0, \ e_2^{(3)} = 0, \ e_3^{(3)} = +\frac{b_3}{\varepsilon}, \ e_1^{(3)} = (-\sqrt{2}\,\frac{b_3}{\varepsilon})/\varepsilon$$

and

$$e_4^{(4)} = -\frac{b_4}{1-\varepsilon}, \ e_3^{(4)} = 0 = e_2^{(4)} = e_1^{(4)} \qquad (38)$$

Consequently,

$$y_1 = \left(\frac{b_1}{\varepsilon} + \frac{b_2(2-\varepsilon)}{\varepsilon^3} - \frac{\sqrt{2}\, b_3}{\varepsilon^2}\right)u_1 + \left(\frac{b_2}{\varepsilon}\right)u_2 + \left(\frac{-\sqrt{2}\, b_2}{\varepsilon^2} + \frac{b_3}{\varepsilon}\right)u_3 - \frac{b_4}{1-\varepsilon}\, u_4. \qquad (39)$$

It is clear from (39) that unless b_2 and b_3 are simultaneously zero, the vector $x_1 = (\varepsilon^3/2b_2)y_1$ is already a good approximation to u_1 if λ is a good approximation to $\lambda_1=-1$. It is interesting to note that if λ is chosen close to $\lambda_4=-2$, then $1-\varepsilon=1+1+\lambda=2+\lambda$ so that $\varepsilon' = 1-\varepsilon$ is a small quantity now while ε is near unity. Then (39) implies that $(\varepsilon'/b_4)y_1$ is now a good approximation to u_4.

Unlike the case of diagonalizable matrices, in defective systems further iterations may bring very slow improvement if any in the eigenvector. E.g., in

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

the eigenvalues are 1, 1 and the unique eigenvector is $u_1=\{1,0\}$. Taking $x_0 = \{a,b\}$ and $\varepsilon=1-\lambda$ it is seen that

$$\begin{bmatrix} \varepsilon & 1 \\ 0 & \varepsilon \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

yields $x_1 = (a\varepsilon-b)/\varepsilon^2$, $y_1 = b/\varepsilon$. When normalized one obtains $\{a\varepsilon-b, b\varepsilon\}$ or approximately $\{1, -\varepsilon\}$. If the process is repeated,

$$\begin{bmatrix} \varepsilon & 1 \\ 0 & \varepsilon \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} a\varepsilon-b \\ b\varepsilon \end{bmatrix}$$

leads to the second approximation $\{1, -\varepsilon/2\}$ to the eigenvector $\{1, 0\}$. Thus when ε is small (as it will be for good eigenvalue approximations), the improvement is insignificant even when one neglects round off errors.

The question now is, given that λ_1 is, say, a double root, how does one determine if there is another eigenvector of λ_1? If there is one, how does one find it? It is known that there is no simple practical answer to the first question. (In fact, independence in itself is a troublesome concept in the numerical sense.) Several theoretical answers are available none of which is of much use in practice. Therefore, proceed under the assumption that there is one more independent vector and try to find it. The method popularized (by Wilkinson and others) was to perturb the approximate eigenvalue λ of λ_1 by a small quantity of the order of 2^{-t}, where t is the number of binary digits carried by the computer used, and go through the calculation again [4]. An alternative procedure suggested by Ortega [5] is used, where one generates a new initial vector x_0' in (24). This approach is appealing because this leaves the matrix in (24) unaltered, as opposed to Wilkinson's method, resulting at least in less programming effort and possibly some saving in computer time. The vector x_0' is produced from a random number generator subroutine. The probability that the second initial vector is identical to the first initial vector is near zero and therefore the probability is very high that the second eigenvector is independent of the first. This is easily seen from the fact that the eigenspace of λ_1 is now a plane and after accepting u_1 as the first vector all one needs for u_2 to be acceptable is that it be in the plane and separated from u_1 by an angle of at least a few degrees. E.g., if 4° separation is sufficient for accuracy in the later orthonormalizing process, the probability is over .95. Of course, if λ_1 does not have a second independent eigenvector (matrix is nondiagonalizable) then every choice of x_0 will lead to the same eigenvector. In case a second attempt leads to the same eigenvector, how does one decide whether this is a failure or this is due to the fact that there are no other independent vectors? It was decided that the reasonable thing to do is to repeat the process with new random vectors a "sufficient" number of times to be "reasonably certain". This number in our program is a parameter and depending on how conservative/optimistic the user is the value could be high/low. It is now clear that the above approach applies just as well to higher multiplicities.

The question of how one determines independence of two (or more vectors is yet to be settled. Once again the theoretical definition of independence is discarded as unsatisfactory. Instead, as soon as a vector is obtained one tests to see if it is a candidate towards producing an orthonormal set of eigenvectors because one may not know, a priori, the dimension of this eigenspace and again one cannot say how many linearly independent eigenvectors can be extracted. Using the Gram-Schmidt process[3] one simultaneously orthonormalizes and tests for independence.

[3] See below.

Recall the earlier comments that process (30) is used with a random initial vector x_0 to obtain an eigenvector u_1 which is normalized to unit length (i.e., $||\cdot||=1$). Process (30) is repeated with a new random vector for x_0 and orthonormalization is done using Gram-Schmidt process to obtain u_2. Even though it is perhaps known to most readers it is necessary to introduce briefly the Gram-Schmidt process to explain when and why it may fail. In the notation used in the next paragraph, the symbols have meaning independent of what they may stand for elsewhere.

Let $\alpha_1, \alpha_2, \ldots, \alpha_n$ be a linearly independent set of vectors and let

$$\beta_1 = \alpha_1/||\alpha_1||$$

$$\mu_2 = \alpha_2 - (\alpha_2, \beta_1)\ \beta_1,\ \beta_2 = \mu_2/||\mu_2||$$

and in general (i.e., for k=2, 3, ..., n)

$$\mu_k = \alpha_k - \sum_{j=1}^{k-1} (\alpha_k, \beta_j)\ \beta_j,\ \beta_k = \mu_k/||\mu_k|| \ .$$

Then $\beta_1, \beta_2, \ldots, \beta_n$ is an orthonormal set spanning the same vector space as that of $\alpha_1, \ldots, \alpha_n$. This is the Gram-Schmidt process. Look at β_2. μ_2 is nonzero because α_2 and β_1 are linearly independent and hence $\alpha_2 \neq (\alpha_2, \beta_1)\beta_1$. But what if α_2 is very nearly equal to $(\alpha_2, \beta_1)\beta_1$? There may be severe cancellation in the computation of μ_2 and consequently the β_2 may be grossly in error. (This kind of cancellation is even more disastrous in later stages.) It is now clear that it is of great advantage, in case severe cancellation is suspected, if one has the choice of replacing the α_2 by another vector and recomputing β_2, rather than continue with a poorly obtained β_2. This is exactly what is incorporated in the program.

By the above process, then, one comes to a point where one has either as many orthonormal eigenvectors as the multiplicity of the eigenvalue at hand or the Gram-Schmidt process fails repeatedly to produce any further eigenvectors. The latter case indicates the probability of nonlinear divisors. In either case it is concluded that all the eigenvectors have been extracted.

4. The Eigenvectors of A

Having obtained the eigenvectors for an eigenvalue λ of the Hessenberg form the next major step in the program is to obtain the corresponding eigenvectors for the original matrix A. It is recalled that

$$H = P_1 P_2 \dots P_{n-2} A P_{n-2} P_{n-1} \dots P_1$$

$$= P^{-1} A P \text{ (say)}$$

where each P_i is (unitary) orthogonal. Then

$$Hx = \lambda x$$

$$P^{-1} A Px = \lambda x$$

$$A(Px) = \lambda (Px) \quad .$$

Hence if x is an eigenvector of H then Px is the corresponding eigenvector of A. Since all the information needed to construct P is saved when computing H from A, Px is readily obtained.

Note further than if x_1, x_2 are two orthonormal eigenvectors of the same eigenvalue λ of H then

$$(x_1, x_2) = 0$$

and

$$
\begin{aligned}
(Px_1, Px_2) &= (x_1, P^T Px_2) \\
&= (x_1, P_1^T P_2^T \ldots (P_{n-2}^T P_{n-2}) \ldots P_1 x_1 \\
&= (x_1, x_2) \\
&= 0
\end{aligned}
$$

Again

$$(Px_1, Px_1) = (x_1, x_1) = 1$$

Thus the eigenvectors obtained by operating with P on the eigenvectors of H corresponding to an eigenvalue automatically form an orthonormal system for A.

5. Examples

In this section five numerical examples are presented. The examples are selected on the basis of features such as multiplicity, defects or ill-conditioning. The exact eigenvalues are known. Thus computed eigenvalues give a measure of accuracy of the Hessenberg form. Also the eigenvectors in most cases can easily be hand-computed for checking. Notice that in general errors in the imaginary parts of eigenvalues as well as eigenvectors are much larger than the errors in the real parts. This is our experience with every difficult example.

In the Hessenberg forms of the print out, the nonzero entries below the subdiagonal are the components of the transformation vectors w (see equation (6)). These are not parts of the Hessenberg matrices, thus should be ignored.[4] Example 1 is the matrix E discussed in page 4. Example 2 has exact eigenvalues $1 \pm 2i$, 4, -1. The third example has 1 as root of multiplicity four. This root has only three distinct eigenvectors. The third and fourth computed eigenvectors correspond to the same exact eigenvector. The errors in the components are of the same order as the corresponding eigenvalue. For example, the third eigenvalue has an $O(10^{-4})$ error in its imaginary part and components of its eigenvector have the same order error. Notice that for the well separated fifth eigenvalue 2, the error in the eigenvector is of order 10^{-19}.

[4] All vectors were obtained after two iterations.

```
     THE 4 -ORDER MATRIX:

 -.10000000+01   .00000000       .00000000       .00000000
  .10000000+01  -.10000000+01    .10000000+01    .10000000+01
  .10000000+01   .00000000      -.10000000+01    .00000000
  .10000000+01   .00000000       .10000000+01   -.20000000+01

HAS HOUSEHOLDER-HESSENBERG FORM:

 -.10000000+01   .00000000       .00000000       .00000000
 -.17320508+01  -.33333333+00    .47140452+00    .81649658+00
  .32505758+00  -.94280904+00   -.16666667+01    .57735027+00
  .32505758+00   .38268343+00    .73725748-17   -.20000000+01


EIGENVALUE(    1  )=  -.10000001+01    .76487064-06
HAS EIGENVECTOR:
  .521987-08   .100215-08         .100000+01  -.256222-06         .253320-06  -.256222-06         .238419-06  -.256222-06

EIGENVALUE(    2  )=  -.99999993+00    .59628325-06
HAS EIGENVECTOR:
  .383640-08  -.366112-09         .100000+01   .175169-06        -.745058-08   .177512-06        -.223517-07   .177512-06

EIGENVALUE(    3  )=  -.99999997+00   -.13611539-05
HAS EIGENVECTOR:
  .302232-08   .103349-08         .100000+01   .424929-06         .109524-05   .424929-06         .108778-05   .424929-06

EIGENVALUE(    4  )=  -.20000000+01    .00000000
HAS EIGENVECTOR:
  .320000-08   .000000            .707107+00   .000000            .372529-08   .000000           -.707107+00   .000000
```

Example 1

```
     THE 4 -ORDER MATRIX:


  .50000000+01  -.20000000+01  -.50000000+01  -.10000000+01
  .10000000+01   .00000000     -.30000000+01   .20000000+01
  .00000000      .20000000+01   .20000000+01  -.30000000+01
  .00000000      .00000000      .10000000+01  -.20000000+01


 HAS HOUSEHOLDER-HESSENBERG FORM:

  .50000000+01   .20000000+01   .50000000+01  -.10000000+01
 -.10000000+01   .00000000     -.30000000+01  -.20000000+01
  .00000000      .20000000+01   .20000000+01   .30000000+01
  .00000000      .00000000     -.10000000+01  -.20000000+01


 EIGENVALUE(    1   )=    .10000000+01   -.20000000+01
 HAS EIGENVECTOR:
   .660225+00    .000000          .233021+00  -.388368+00       .427205+00   .388368+00       .388368-01   .155347+00

 EIGENVALUE(    2   )=    .39999999+01    .14901161-07
 HAS EIGENVECTOR:
   .978668+00    .000000          .159565+00   .000000          .127652+00   .000000          .212754-01   .000000

 EIGENVALUE(    3   )=    .10000000+01    .20000000+01
 HAS EIGENVECTOR:
   .660225+00    .916506-09       .233021+00   .368368+00       .427205+00  -.388368+00       .388368-01  -.155347+00

 EIGENVALUE(    4   )=  -.10000000+01    .00000000
 HAS EIGENVECTOR:
   .577350+00    .000000          .107540-07   .000000          .577350+00   .000000          .577350+00   .000000
```

Example 2

```
    THE 5 -ORDER MATRIX:

  .10000000+01    .00000000      .00000000      .00000000      .00000000
  .00000000       .10000000+01   .00000000      .00000000      .00000000
  .00000000       .10000000+01   .10000000+01   .00000000      .00000000
  .00000000       .00000000      .00000000      .10000000+01   .00000000
  .00000000       .00000000      .00000000      .00000000      .20000000+01


HAS HOUSEHOLDER-HESSENBERG FORM:

  .10000000+01    .00000000      .00000000      .00000000      .00000000
  .00000000       .10000000+01   .00000000      .00000000      .00000000
  .00000000      -.10000000+01   .10000000+01   .00000000      .00000000
  .00000000       .00000000      .00000000      .10000000+01   .00000000
  .00000000       .00000000      .00000000      .00000000      .20000000+01


EIGENVALUE(    1  )=    .10000000+01    .00000000
HAS EIGENVECTOR:
  .142634+00    .000000             .236626-10    .000000            -.326483-11    .000000             .989776+00    .000000
  .104249-19    .000000

EIGENVALUE(    2  )=    .10000000+01    .00000000
HAS EIGENVECTOR:
 -.989776+00    .000000             .335025-10    .000000            -.639435-10    .000000             .142634+00    .000000
  .146828-20    .000000

EIGENVALUE(    3  )=    .99999999+00    .86316746-04
HAS EIGENVECTOR:
  .636646-10    .227374-12          .576652-08   -.431580-04         -.100000+01   -.175345-14          -.119371-10    .454747-12
 -.183376-12    .318400-12

EIGENVALUE(    4  )=    .99999999+00   -.86316746-04
HAS EIGENVECTOR:
 -.121730-04   -.253798-04          .755871-08    .431596-04         -.100000+01    .000000             -.293215-04    .133670-04
  .294198-12   -.252190-13

EIGENVALUE(    5  )=    .20000000+01    .00000000
HAS EIGENVECTOR:
  .155881-19    .000000             .689390-20    .000000             .239782-19    .000000             .783081-20    .000000
  .100000+01    .000000
```

Example 3

```
THE 6 -ORDER MATRIX:

 .10000000+02  -.19000000+02   .17000000+02  -.12000000+02   .40000000+01   .10000000+01
 .90000000+01  -.18000000+02   .17000000+02  -.12000000+02   .40000000+01   .10000000+01
 .80000000+01  -.16000000+02   .15000000+02  -.11000000+02   .40000000+01   .10000000+01
 .60000000+01  -.12000000+02   .12000000+02  -.10000000+02   .40000000+01   .10000000+01
 .40000000+01  -.80000000+01   .80000000+01  -.60000000+01   .10000000+01   .20000000+01
 .20000000+01  -.40000000+01   .40000000+01  -.30000000+01   .10000000+01   .00000000


HAS HOUSEHOLDER-HESSENBERG FORM:

 .10000000+02   .62775760+01  -.19432585+02   .11167909+02  -.16382649+02  -.92373849+00
-.14177447+02  -.89552238+01   .29339944+02  -.17600451+02   .27363024+02   .11857366+01
 .31206364+00  -.54499757-01  -.44776119-01  -.65384122-02   .12783003+01  -.22133688+00
 .23404773+00   .29443222+00   .17064612-15  -.57283070+00   .95519867+00  -.10207051+01
 .15603182+00   .19628815+00   .45227553+00  -.18937292+00  -.14234266+01  -.10073645+01
 .78015910-01   .98144073-01   .22687437+00   .23132464+00   .83565140-07  -.10037427+01


EIGENVALUE(    1  )=  -.74505806-08  -.10000000+01
HAS EIGENVECTOR:
 -.536172+00  -.353908-01      -.536171+00  -.353908-01      -.474237+00   .212345-01      -.355678+00   .159259-01
 -.237119+00   .106173-01      -.118559+00   .530863-02

EIGENVALUE(    2  )=   .99999999+00   .00000000
HAS EIGENVECTOR:
 -.628971+00   .000000         -.524142+00   .000000         -.419314+00   .000000         -.314485+00   .000000
 -.209657+00   .000000         -.104828+00   .000000

EIGENVALUE(    3  )=   .43156945-15   .10000000+01
HAS EIGENVECTOR:
 -.536172+00   .353908-01      -.536171+00   .353908-01      -.474237+00  -.212345-01      -.355678+00  -.159259-01
 -.237119+00  -.106173-01      -.118559+00  -.530863-02

EIGENVALUE(    4  )=  -.10000004+01  -.11805990-05
HAS EIGENVECTOR:
 -.468521+00  -.233390-07      -.468521+00  -.233390-07      -.468521+00  -.233391-07      -.468521+00  -.233391-07
 -.312347+00   .107360-06      -.156174+00   .536798-07

EIGENVALUE(    5  )=  -.99999955+00  -.35469247-06
HAS EIGENVECTOR:
 -.468521+00  -.701185-08      -.468521+00  -.701186-08      -.468521+00  -.701187-08      -.468521+00  -.701187-08
 -.312348+00   .322546-07      -.156174+00   .161273-07

EIGENVALUE(    6  )=  -.10000001+01   .15352919-05
HAS EIGENVECTOR:
 -.468521+00   .303509-07      -.468521+00   .303509-07      -.468521+00   .303510-07      -.468521+00   .303509-07
 -.312348+00  -.139614-06      -.156174+00  -.698071-07
```

Example 4

```
   THE 8 -ORDER MATRIX:

  .00000000      .40000000+01   .80000000+01   .80000000+01   .00000000      .30000000+01   .60000000+01   .60000000+01
  .16000000+02   .44000000+02   .00000000      .44000000+02   .12000000+02   .33000000+02   .00000000      .33000000+02
  .12000000+02   .28000000+02  -.28000000+02   .80000000+01   .90000000+01   .21000000+02  -.21000000+02   .60000000+01
 -.16000000+02  -.36000000+02   .28000000+02  -.16000000+02  -.12000000+02  -.27000000+02   .21000000+02  -.12000000+02
  .00000000     -.20000000+01  -.40000000+01  -.40000000+01   .00000000     -.10000000+01  -.20000000+01  -.20000000+01
 -.80000000+01  -.22000000+02   .00000000     -.22000000+02  -.40000000+01  -.11000000+02   .00000000     -.11000000+02
 -.60000000+01  -.14000000+02   .14000000+02  -.40000000+01  -.30000000+01  -.70000000+01   .70000000+01  -.20000000+01
  .80000000+01   .16000000+02  -.14000000+02   .80000000+01   .40000000+01   .90000000+01  -.70000000+01   .40000000+01


HAS HOUSEHOLDER-HESSENBERG FORM:

  .00000000     -.69843029+00   .55659886+00   .53507195+00  -.46767697+00  -.17710448+01   .79184061+00   .14830164+02
 -.28635642+02  -.44390244+00  -.13772015+02  -.91623260+00   .34515084+02   .17757719+02  -.10074719+03  -.29665505+02
  .23734052+00  -.55701678+00  -.31052658+00  -.15232598+01  -.50597801+00  -.51733452+01  -.25643604+01   .35472972+02
 -.31645403+00  -.20303159+00  -.52337030+01   .18543513+00   .26123368+01   .34291409+00   .66270771+01   .21585576+01
  .00000000      .44757694+00  -.60778441-01   .19158788+01   .44179437+00   .33931676+01   .41333168+01  -.29926103+02
 -.15822701+00   .84974597-01  -.37442015+00  -.82850971-01   .24268550+00   .19212855+00  -.36768091+01  -.27730217+01
 -.11867026+00  -.16005752+00  -.32971335+00   .16246764+00  -.82963807-01   .20641637+01  -.13283130+01   .95478346+01
  .15822701+00   .13881387+00   .34453093+00  -.14606182+00  -.56796067+00  -.56667550+00   .15101421+01   .12633839+01


EIGENVALUE(    1  )=  -.60000000+01    .00000000
HAS EIGENVECTOR:
  .892052-01   .000000         -.446026+00   .000000          -.446026+00   .000000          .535231+00   .000000
 -.594701-01   .000000          .297351+00   .000000           .297351+00   .000000         -.356821+00   .000000

EIGENVALUE(    2  )=   .60000000+01    .00000000
HAS EIGENVECTOR:
 -.116510+00   .000000         -.582552+00   .000000          -.349531+00   .000000          .466041+00   .000000
  .776736-01   .000000          .388368+00   .000000           .233021+00   .000000         -.310694+00   .000000

EIGENVALUE(    3  )=  -.30000000+01    .25212643-14
HAS EIGENVECTOR:
  .758098-01  -.666135-17      -.379049+00   .397397-16       -.379049+00  -.711214-17       .454859+00  -.449527-17
 -.758098-01  -.132991-16       .379049+00   .115009-16        .379049+00   .359666-16      -.454859+00  -.393258-16

EIGENVALUE(    4  )=  -.20000000+01   -.27451086-14
HAS EIGENVECTOR:
  .217818+00   .724823-15      -.523983+00  -.464047-15       -.342286+00  -.131350-15       .497871+00   .348292-15
 -.145212+00  -.296008-15       .352653+00   .369219-15        .228191+00   .859310-16      -.331914+00  -.299090-15

EIGENVALUE(    5  )=   .30000000+01    .34696242-17
HAS EIGENVECTOR:
 -.990148-01  -.155190-16      -.495074+00   .356423-16       -.297044+00   .163442-16       .396059+00  -.404949-16
  .990148-01   .906438-17       .495074+00   .771895-16        .297044+00   .530381-16      -.396059+00  -.598225-16

EIGENVALUE(    6  )=   .20000000+01   -.63598853-17
HAS EIGENVECTOR:
 -.202045+00  -.120426-16      -.490680+00   .122613-17       -.375226+00  -.950343-17       .519543+00   .157425-16
  .134696+00  -.534860-18       .327120+00  -.263910-16        .250150+00  -.895412-17      -.346362+00   .101814-16

EIGENVALUE(    7  )=  -.10000000+01    .27626673-14
HAS EIGENVECTOR:
  .185110+00   .492087-15      -.449553+00  -.262321-15       -.290867+00  -.226667-16       .423109+00   .117440-15
 -.185110+00  -.992879-16       .449553+00  -.244487-15        .290987+00  -.279736-15      -.423109+00   .339233-15

EIGENVALUE(    8  )=   .99999999+00    .63154101-18
HAS EIGENVECTOR:
 -.171705+00  -.704524-16      -.416997+00   .386931-16       -.318880+00   .293441-16       .441527+00  -.404114-16
  .171705+00   .241141-16       .416997+00   .827350-16        .318880+00   .551271-16      -.441527+00  -.781448-16
```

Example 5

Example 4 is also a defective matrix and again exhibits the larger errors in the imaginary parts. Notice that the last three vectors in this example are almost identical in the significant components.

In contrast to the first four examples, the last one is a matrix with well-separated eigenvalues ±1, ±2, ±3, ±6. As expected the results are excellent for this example.

Acknowledgement

I am grateful to L. D. Nelson for careful reading of an earlier draft of this memorandum, for pointing out several errors and for suggesting several changes. I must express my sincere thanks to R. R. Singers who performed all the programming, experimenting and testing with infinite patience.

V. Thuraisamy

1033-VT-jf V. Thuraisamy

Attachments
References
Appendix

REFERENCES

1. P. A. Businger, Reducing a Matrix to Hessenberg Form, Math. Comp. 23 (1969) pp. 819-821.

2. J. H. Wilkinson, Householder's Method for the Solution of the Algebraic Eigenproblem, Computer Journal 3 (1960) pp. 23-27.

3. R. S. Martin, et. al., The Q-R Algorithm for Real Hessenberg Matrices, Numer. Math. 14 (1970) pp. 219-231.

4. J. H. Wilkinson, The Algebraic Eigenvalue Problem, Oxford (1965).

5. J. Ortega, The Given's Householder Method for Symmetric Matrices, Math. Methods for Digital Computers, Vol. 2, (Editors, Ralston and Wilf) Wiley (1966).

6. Hoffman and Kunze, Linear Algebra, Prentice Hall (1961).

APPENDIX

Here a proof is indicated for the convergence of the Wielandt iteration process when applied to defective matrices. In the text a specific example was considered. Essentially the same argument would suffice in general. Consequently, manifestly repetitive steps are avoided.

Let λ_1 be an eigenvalue of multiplicity s_1 and let the dimension of the eigenspace of λ_1 be s_1' with $s_1' < s_1$. Then take $u_1, u_2, \ldots, u_{s_1'}$, a set of linearly independent eigenvectors of λ_1, and the additional vectors $u_{s_1'+1}, \ldots, u_{s_1}$ so that together these s_1 vectors form an orthonormal basis for W_1 of Theorem 1. This is done for each of the W_i. Concentrate on the subspace W_1. (The case when $s_1'=s_1$ is easily disposed of.)

With this definition of the u_i; $i=1,2,\ldots,n$; one can again write down equations (23) - (26). Instead of (27) equations of the following type are obtained:

$$b_1u_1 = \sum_{k=1}^{s_1'} (\lambda_k-\lambda)\, e_k^{(1)}u_k + \sum_{k=s_1'+1}^{n} (H-\lambda I)e_k^{(1)}u_k \, . \qquad \text{(A1)}$$

The aim is to evaluate $e_k^{(i)}$, $k=1,\ldots,n$, $i=1,2,\ldots,n$. From the Primary Decomposition Theorem it is known that

$$Hu_k = \sum_{\ell=1}^{s_1} \beta_{k\ell}\, u_\ell \, , \quad k=s_1'+1, \ldots, s_1 \qquad \text{(A2)}$$

and

$$Hu_k = \sum_{\ell=s_1+1}^{n} \beta_{k\ell}\, u_\ell \quad , k=s_1+1,\ldots,n \qquad \text{(A3)}$$

for scalars $\beta_{k\ell}$. Substituting (A2), (A3) in (A1) and using linear independence of u_i one obtains

$$b_1 = (\lambda_1-\lambda)e_1^{(1)} + \sum_{k=s_1'+1}^{s_1} \beta_{k1}\, e_k^{(1)} \tag{A4}$$

$$0 = (\lambda_i-\lambda)e_i^{(1)} + \sum_{k=s_1'+1}^{s_1} \beta_{ki}e_k^{(1)} \quad , \quad i=2,\ldots,s_1' \tag{A5}$$

$$0 = \sum_{k=s_1'+1}^{s_1} \beta_{ki}e_k^{(1)} - \lambda e_i^{(1)} \quad , \quad i=s_1'+1,\ldots,s_1 \tag{A6}$$

and

$$0 = \sum_{s_1+1}^{n} e_k^{(1)}\, \beta_{ki} - \lambda e_i^{(1)} \quad , \quad i=s_1+1,\ldots,n. \tag{A7}$$

If the last four equations are written out as a linear system for the unknowns $e_k^{(1)}$, k=1,...,n, one obtains a matrix of the form

$$\left[\begin{array}{c|c|c} D & B_1 & 0 \\ \hline 0 & B_2 & 0 \\ \hline 0 & 0 & B_3 \end{array}\right] \tag{A8}$$

D is an s_1' order diagonal matrix with $d_{ii} = \lambda_i - \lambda$. B_1 is $s_1' \times (s_1 - s_1')$ and B_2, B_3 are square matrices.

$$B_2 = \begin{bmatrix} \beta_{s_1'+1,s_1'+1}-\lambda & \beta_{s_1'+2,s_1'+1} & \cdots & \beta_{s_1,s_1'+1} \\ \beta_{s_1'+1,s_1'+2} & \beta_{s_1'+2,s_1'+2}-\lambda & & \beta_{s_1,s_1'+2} \\ \vdots & & & \\ \beta_{s_1'+1,s_1} & \cdots\cdots & & \beta_{s_1,s_1}-\lambda \end{bmatrix} \qquad \text{(A9)}$$

and is such that

$$B_2 \begin{bmatrix} e^{(1)}_{s_1'+1} \\ e^{(1)}_{s_i'+2} \\ \vdots \\ e^{(1)}_{s_1} \end{bmatrix} = 0 \qquad \text{(A10)}$$

Since λ is arbitrary, it may be assumed that B_2 is nonsingular. Then $e_k^{(1)}=0$, $k=s_1'+1, \ldots, s_1$. By the same argument one will also have $e_k^{(1)}=0$, $k=s_1+1, \ldots, n$. Hence from the first s_1' equations it is concluded that

$$e_1{}^{(1)} = b_1/(\lambda_1-\lambda)$$

$$e_k{}^{(1)} = 0 \quad , k=2,\ldots,n.$$

It is now clear that starting from an equation of type (A1) one finds $e_k^{(i)}$, $i=1,2,\ldots,s_1'$ to be

$$\begin{aligned} e_i^{(i)} &= b_i/(\lambda_i-\lambda) \\ e_k^{(i)} &= 0 \quad , \ k \neq i. \end{aligned} \qquad \text{(A11)}$$

For $s_1' < i \leq s_1$ (A8) leads to

$$D \begin{bmatrix} e_i^{(i)} \\ \vdots \\ e_{s_1'}^{(i)} \end{bmatrix} + B_1 \begin{bmatrix} e_{s_1'+1}^{(i)} \\ \vdots \\ e_{s_1} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

Noting that D^{-1} is a diagonal matrix with nonzero elements $d^*_{ii} = (\lambda_i-\lambda)^{-1}$, one has

$$\begin{bmatrix} e_1^{(i)} \\ \vdots \\ e_{s_1'}^{(i)} \end{bmatrix} = -D^{-1}B_1 \begin{bmatrix} e_{s_1'+1}^{(i)} \\ \vdots \\ e_{s_1}^{(i)} \end{bmatrix} = -D^{-1}B_1B_2^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_i \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \text{(A12)}$$

and

$$e_k^{(i)} = B_2^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_i \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad , \ k = s_1'+1, \ \ldots, s_1.$$

It still follows that $e_k^{(i)}=0$, $k=s_1+1,\ldots,n$.

Finally, for $i>s_1$, one has

$$e_k^{(i)} = 0 \ , \ k=1,\ldots,s_1$$

and

$$\begin{bmatrix} e_{s_1+1}^{(1)} \\ \cdot \\ \cdot \\ \cdot \\ e_n \end{bmatrix} = B_3^{-1} \begin{bmatrix} 0 \\ \vdots \\ b_i \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{A13}$$

Thus one may write

$$e_k^{(i)} = O(b_i \cdot ||B_3^{-1}||) \ . \tag{A14}$$

Without going into details, it is seen how relative separation between distinct eigenvalues plays an important role in the rate of convergence (c.f., (38), (39)). In fact it is pertinent to write

$$||B_3^{-1}|| = O(\min_{j\geqslant s_1} |\lambda_1-\lambda_j|^{-1}) \tag{A15}$$

With the help of (A11) - (A15) it can now be claimed that after one step of the iteration process

$$Hy_1 = x_0$$

$$x_0 = y_1/||y_1|| \quad ,$$

x_1 has the form

$$x_1 = \sum_{k=1}^{s_1'} O(b_k')u_k + \sum_{k=s_1'+1}^{s_1} O(b_k')(\lambda_1-\lambda)u_k$$

$$+ \sum_{k=s_1+1}^{n} \frac{O(b_k'')|\lambda_1-\lambda|}{\min\{|\lambda_1-\lambda_j|,\ j=s_1+1,\ldots,n\}}$$

Here b_k' is a function f_k with arguments $b_1, b_2, \ldots, b_{s_1}$. That is, $b_k' = f_k(b_1,\ldots,b_{s_1})$ and similarly $b_k'' = f_k^*(b_{s_1+1},\ldots,b_n)$. Without further ado write in general for s=1,2,3,...

$$x_s = \sum_{k=1}^{s_1'} \tilde{b}_k^{(s)} u_k + \sum_{k=s_1'+1}^{s_1} \tilde{b}_k^{(s)} |\lambda_1-\lambda|^s u_k$$

$$+ \frac{1}{\min\limits_{j=s_1+1,\ldots,n} \{|\lambda_1-\lambda_j|\}} \sum_{k=s_1+1}^{n} \tilde{b}_k^{(s)} |\lambda_1-\lambda|^s$$

where the numbers $\tilde{b}_k^{(s)}$ depend only on the components of the initial vector x_0. Thus convergence is established. Note that if λ is a good approximation to λ_1, then x_1 is approximately in the eigenspace of λ_1. If λ_1 is well separated from the other distinct eigenvalues, x_1 is already a good eigenvector.